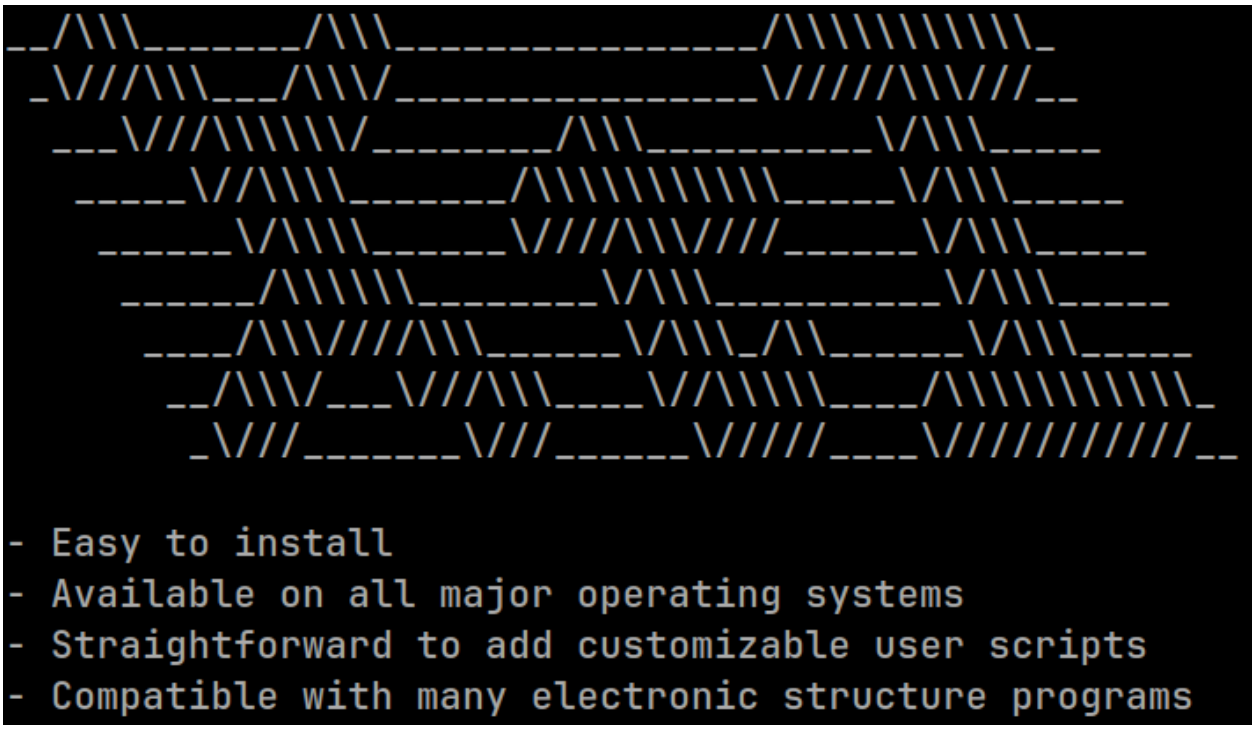
XYZ to Input: A Command-line Utility for Rapid Input File and Job Submission Script Generation

Nicholas J. Roberts, Lawrence L. Matsuoka

Department of Chemistry and the Centre for Advanced Materials and Related Technology, University of Victoria, Victoria, British Columbia, V8P 5C2, Canada

E-mail:njeffrob@tutanota.com



1 Abstract

Consider publishing in Organometallics, Digital Discovery or JChemEd as other relevant journals - this is meant to be a short comm. The story: a useful tool to experimentalists/theorists doing molecular structure DFT to rapidly generate input files

2 Introduction

Studying molecular species using quantum mechanics is a flourishing area of research. In the modern day, there are a plethora of methods available to probe molecular/macromolecular properties: *ab initio*[1], quantum Monte-Carlo,[2] semiempirical[3], hybrid quantum/molecular mechanical[4], and long–range perturbation theory[5] all have their own strengths/weaknesses and can be applied successfully to chemical systems. However, none of these quantum methods are as widely studied in chemistry as density-functional theory (DFT).[6] Indeed, among the top 100 most cited papers of all time, 12 of them are accredited to DFT.[7] There are numerous DFT programs to choose from,[8–19] all with their own unique features and applications. With many new avenues of research in DFT, it is an exciting time to be a theoretical chemist.

While DFT is a bustling field of research, it is also a practical tool used alongside experimental methods to provide greater insight into molecular problems. DFT is commonly employed to calculate reaction pathways,[20–22] predict spectral properties,[23–25] calculate molecular orbital properties,[26] and much, much more. One would be hard pressed to find a modern synthetic paper that does not include DFT of some kind in it. In fact, DFT is such a commonly employed tool for experimental chemists that many traditionally "synthetic" journals now publish articles that are purely based on DFT calculations.[27–44] One could go so far as to say that DFT could be considered as crucial as many forms of spectroscopy to modern experimental chemistry.

Despite the ubiquity of molecular DFT in many fields of chemistry, the manual work that is required to obtain these results is often cumbersome. Popular electronic structure programs, such as Gaussian,[9] ORCA,[10,45] or FHI-AIMS,[8] are often not run on a users computer but on a high-performance supercomputer. This (generally) means that a users time spent working on DFT calculations is spent looking at a terminal. Generating the files to perform these calculations is often tedious; while there are graphical ways to generate input files, such as Avogadro,[46] Gaussview,[47] or Gabedit,[48] these require the user to download output files onto their own systems, open the files, and generate the input. The user then has to upload the new structure to the cluster, along with a new submission script file to submit this input file as a job. This process is needlessly inefficient, and slows down what should be an automated process. A program that comes close to automating file input generation is critic2,[49,50] which allows users to convert between file formats via the command line, but this still requires manually editing the input file. And while it may be the case that users have developed their own solutions to this, a more universal tool would be of benefit to novice computational chemists or those who do theoretical chemistry on top of experimental work.

Herein we report XYZ to Input (XtI), a command line tool designed to rapidly generate input files for molecular DFT calculations. The program is available on all major operating systems (Linux, MacOS and Windows), with a strong emphasis on ease of use and portability. The program is available on Debian, RHEL and Arch package managers, as well as homebrew for MacOS and WinGet for Windows. The raw source code can also be compiled either manually or via makefile. In the most stringent of cases, the main program, written in Lua, can be run using the Lua interpreter, obviating the need to install anything other than the interpreter and the scripts. It is the authors hope that this program saves time for researchers performing these manually intensive calculations.

Things I want this program to have:

- easy installation (either C/Rust compiling OR add it to Linux repositories; need to see how to embed Lua into C/Rust or how to add a lua script to repositories)
- should be available on: debian/RHEL/Arch based distros, homebrew, Mac (maybe via homebrew), and windows (ideally via WSL but we'll see)
- options for:
  - xti –options (pure command line tool like grep)
- API to let users add custom modifications via Lua (probably gonna take some time to learn to do, but likely worth it)

3 Functions

The original XtI program, consisting of a few Lua files, was designed to convert XYZ coordinate files into ORCA input files to speed up job submissions. When scaling this up to be more broadly applicable, we wanted the program to remain straightforward to both use and modify to suit a variety of user needs. To this end, a few things were considered:

1. ease of use of the program;
2. a list of relevant quantum chemistry packages;
3. file conversion options; and
4. submission script generation.

The original program, while functional, left a lot to be desired; Lua files had to be run manually in a single directory, meaning there was much copy/pasting of files from one location to another. This was faster than previous workflows, but still largely inefficient. To improve upon this, the main program loop was re-written in C as a command-line tool. C, while cumbersome to write code for, has the unique advantage in that Lua can be directly embedded into C code. This means that the challenging bits, such as file manipulation, can be handled in Lua, and the C code can simply act as the interface between the user and the program. In addition, as the main program is written in C, it can be compiled on any machine and can therefore be utilized on any operating system.

The following subsections describe the primary functions of XtI.

3.1 Converting XYZ Files to Input Files

As the name suggests, the main feature of XtI is the ability to convert XYZ atomic coordinate files into input files for quantum chemistry programs. A list of supported programs can be seen in Table 1.

Table 1: List of quantum chemistry programs that XtI supports for the generation of input/output files.

While this list doesn't cover every single quantum chemistry package, it does cover the most popular ones found in the literature (with a few more for good measure). In addition, users can add additional programs with relative ease thanks to the programs extensibility (see section 4.4).

3.2 Converting Output Files to XYZ Files

While some programs, such as ORCA, generate XYZ coordinate files as part of the standard output, many do not. As the XYZ file format is both commonly used to share molecular geometries and is crucial to form input files from XtI, a method to convert output formats to XYZ formats is also included in the program. This output to xyz feature supports the same programs as shown in Table 1.

With respect to extensibility, adding new quantum program options is challenging. While still possible, and some examples for how to do so are provided, the extraction of coordinates from output files is very program dependent. For this reason, XtI cannot claim to be readily extended by inexperienced users. However, for more experienced programmers, the source code is still readily modifiable and the examples provided should provide a good starting point to adding additional functionality.

3.3 Generating Submission Script Files

As mentioned in the introduction, many users do not run calculations locally but perform them on high-performance clusters (HPCs). This means that, on top of users having to modify input files of their molecular geometries, they also need to create a submission script. While script templates are often provided by the HPC providers, they are still required for every job a user submits. To account for this, XtI has an option that automatically generates a generic submission script along with the input file. While some manual editing of the source files is required for this functionality to work (as every HPC has different submission script requirements), once set up this saves the user an enormous amount of time as compared to copy/pasting submission script files.

4 Options, Usage and Extension

4.1 Options

XtI has a number of straightforward options, as can be seen in Table 2 below.

Option	Description
-p	Specifies the electronic structure program for input/output.
-j	Specifies the job type when generating an input file. The default is single point ("sp").
-i	Specifies generation of an input file from an XYZ file.
-o	Specifies generation of an XYZ file from an output file.
-s	Generates a job submission script for submitting calculations to a scheduler.
-h	Displays a help message on how to use the program.

Table 2: List of options available for the XtI program.

The options included are meant to be straightforward for users to understand and use. With only five real command line options (with '-h' only displaying a help message), XtI is a compact and simple tool.

4.2 Usage

One can use XtI according to Scheme 1:

```
xti [options] [program] [file]
```
(1)

The usage of XtI is heavily inspired by that of command line tools such as `grep`; it's meant to be as painless to use as possible. This means it is possible to combine options into a single command, e.g.

```
xti -psi orca example.xyz
```
(2)

This ease of use in the terminal makes XtI feel as comfortable to use as tools such as `man`, `cat`, `tail`, etc. This is crucial to the design of XtI: it is meant to easily slot into existing user computational workflows.

Let's go through an example of where XtI can be incorporated into user workflow. A user has just ran an optimization in Gaussian16 for the structure of a generic molecule called simply `'molecule'`. Their directory may look something like this:

```
/
└── home
    └── scratch
        └── molecule
            └── optimization
                ├── molecule.gjf
                ├── molecule.log
                └── molecule.sh
```

From the .log file, the user has found that the optimization has converged; always a pleasant sight. The user would now like to generate two new input files: one for a frequency calculation (to confirm a true minimum has been found), and one for a single point calculation at a higher level of theory (for more accurate energy calculations). To obtain these input files, the user can perform the following commands:

```
xti -po gaussian molecule.log
```
(3)

```
xti -psij gaussian freq molecule.log
```
(4)

```
xti -psij gaussian sp molecule.log
```
(5)

Scheme 3 generates an XYZ coordinate file from molecule.log. Scheme 4 generates a gaussian input file and job submission script for a frequency calculation, and 5 does the same but for a single point calculation. Their directory will now look like the following:

```
/
└── home
    └── scratch
        └── molecule
            └── optimization
                ├── molecule.gjf
                ├── molecule.log
                ├── molecule.sh
                ├── molecule.xyz
                ├── molecule-freq.gjf
                ├── molecule-freq.sh
                ├── molecule-sp.gjf
                └── molecule-sp.sh
```

Note that XtI will not create new directories for the input files/submission scripts, so users who like to separate their jobs by directory will need to perform this manually. Aside from debating about file management best practices, in three short commands xti allowed the user to generate two input files and two submission scripts for jobs they needed to run, all without having to open any files graphically or manually extract XYZ coordinates from the output file. This example illustrates xti's simple but powerful usage.

For a video tutorial of this example, please see the Supporting Information.

4.3 Editing the Program Files

In our example above, two different kind of input files were generated, but how are we sure that those input files contain the proper keywords/methods that we want our calculations to have? The answer: we define this in the source code.

At the very top of the `inp_maker` and `sh_maker` Lua files, there are a set of predefined constants that the user can modify to their liking. The convenient part of this is that the user need only define their methods once, and all subsequent calculations will use these methods.

XtI, however, does not provide all options that a user may need for certain programs, e.g. solvent correction. This leads into the next convenient feature of XtI: its extensibility.

4.4 Extensibility

One of the highlights of XtI is that the code can be readily modified by relatively inexperienced users. While the main command-line logic was written in C, a historically beginner unfriendly language, the C code itself is actually pretty minimal. This is because most of what the C code does is call Lua files to perform its operations. Lua, in contrast to C, is very beginner friendly, with code that is straightforward to read/write.

In addition, the inclusion of a `make` file ensures that it is easy to build XtI on any platform.

5 Conclusions

The rapid generation of input files from a simple command-line utility has been developed. XtI provides useful functionality to both experienced and novice theoreticians, allowing for improved workflows without the need for learning a complex tool. It is the hope of the authors that XtI helps save researchers time and energy when working on lengthy DFT projects.

Acknowledgements

NJR thanks NSERC for funding.

References

[1] Friesner, R. A. Ab initio quantum chemistry: Methodology and applications. *Proceedings of the National Academy of Sciences* **2005**, *102*, 6648–6653, DOI: 10.1073/pnas.0408036102, Publisher: Proceedings of the National Academy of Sciences.

[2] Dubecky, M. QUANTUM MONTE CARLO FOR NONCOVALENT INTERACTIONS: A TUTORIAL REVIEW. *acta physica slovaca* **2014**, *64*, 501 – 574, DOI: 10.2478/apsrt-2014-0005.

[3] Ramachandran, K. I. et al. , Eds. *Computational Chemistry and Molecular Modeling: Principles and Applications*; Springer: Berlin, Heidelberg, 2008; pp 139–154, DOI: 10.1007/978-3-540-77304-7_7.

[4] Csizi, K.-S.; Reiher, M. Universal QM/MM approaches for general nanoscale applications. *WIREs Computational Molecular Science* **2023**, *13*, e1656, DOI: 10.1002/wcms.1656, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wcms.1656.

[5] Escamilla, L. et al. Perturbation theory for very long-range potentials. *Journal of Molecular Liquids* **2013**, *185*, 20–25, DOI: 10.1016/j.molliq.2012.11.017.

[6] Teale, A. M. et al. DFT exchange: sharing perspectives on the workhorse of quantum chemistry and materials science. *Phys. Chem. Chem. Phys.* **2022**, *24*, 28700–28781, DOI: 10.1039/D2CP02827A, Publisher: The Royal Society of Chemistry.

[7] Van Noorden, R. et al. The top 100 papers. *Nature News* **2014**, *514*, 550, DOI: 10.1038/514550a, Cg_type: Nature News Section: News Feature.

[8] *Ab initio* molecular simulations with numeric atom-centered orbitals. *180*.

[9] Frisch, M. J. et al. Gaussian˜16 Revision C.01. 2016; Gaussian Inc. Wallingford CT.

[10] Neese, F. et al. The ORCA quantum chemistry program package. *The Journal of Chemical Physics* **2020**, *152*, 224108, DOI: 10.1063/5.0004608.

[11] Gonze, X. et al. The Abinit project: Impact, environment and recent developments. *Computer Physics Communications* **2020**, *248*, 107042, DOI: 10.1016/j.cpc.2019.107042.

[12] Clark, S. J. et al. First principles methods using CASTEP. *Zeitschrift für Kristallographie - Crystalline Materials* **2005**, *220*, 567–570, DOI: 10.1524/zkri.220.5.567.65075, Publisher: De Gruyter (O).

[13] Barca, G. M. J. et al. Recent developments in the general atomic and molecular electronic structure system. *The Journal of Chemical Physics* **2020**, *152*, 154102, DOI: 10.1063/5.0005188.

[14] Aprà, E. et al. NWChem: Past, present, and future. *The Journal of Chemical Physics* **2020**, *152*, 184102, DOI: 10.1063/5.0004997.

[15] Shao, Y. et al. Advances in molecular quantum chemistry contained in the Q-Chem 4 program package. *Molecular Physics* **2015**, *113*, 184–215, DOI: 10.1080/00268976.2014.952696, Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/00268976.2014.952696.

[16] Giannozzi, P. et al. QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. *J. Phys.: Condens. Matter* **2009**, *21*, 395502, DOI: 10.1088/0953-8984/21/39/395502.

[17] Balasubramani, S. G. et al. TURBOMOLE: Modular program suite for ab initio quantum-chemical and condensed-matter simulations. *The Journal of Chemical Physics* **2020**, *152*, 184107, DOI: 10.1063/5.0004635.

[18] García, A. et al. Siesta: Recent developments and applications. *The Journal of Chemical Physics* **2020**, *152*, 204108, DOI: 10.1063/5.0005077.

[19] VASP - Vienna Ab initio Simulation Package. https://www.vasp.at/.

[20] Maier, G. et al. Reactions of Silicon Atoms with Nitrogen: A Combined Matrix Spectroscopic and Density Functional Theory Study1. *Organometallics* **2000**, *19*, 4775–4783, DOI: 10.1021/om000234r, Publisher: American Chemical Society.

[21] Nilsson, S. M. E. et al. Mechanism of the Oxidation of Heptafulvenes to Tropones Studied by Online Mass Spectrometry and Density Functional Theory Calculations. *J. Org. Chem.* **2019**, *84*, 13975–13982, DOI: 10.1021/acs.joc.9b02078, Publisher: American Chemical Society.

[22] Zhang, J. et al. CO2 Reforming of Ethanol: Density Functional Theory Calculations, Microkinetic Modeling, and Experimental Studies. *ACS Catal.* **2020**, *10*, 9624–9633, DOI: 10.1021/acscatal.9b05231, Publisher: American Chemical Society.

[23] White, R. E.; Hanusa, T. P. Prediction of 89Y NMR Chemical Shifts in Organometallic Complexes with Density Functional Theory. *Organometallics* **2006**, *25*, 5621–5630, DOI: 10.1021/om060695y, Publisher: American Chemical Society.

[24] Zhang, Y.-P. et al. Predicting Infrared Spectra of Nerve Agents Using Density Functional Theory. *J Appl Spectrosc* **2016**, *83*, 673–681, DOI: 10.1007/s10812-016-0347-5.

[25] Besley, N. A. Modeling of the spectroscopy of core electrons with density functional theory. *WIREs Computational Molecular Science* **2021**, *11*, e1527, DOI: 10.1002/wcms.1527, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wcms.1527.

[26] Guerrin, C. et al. Indolino-Oxazolidine Acido- and Photochromic System Investigated by NMR and Density Functional Theory Calculations. *J. Org. Chem.* **2018**, *83*, 10409–10419, DOI: 10.1021/acs.joc.8b01482, Publisher: American Chemical Society.

[27] Bao, J. L. et al. Density Functional Theory of the Water Splitting Reaction on Fe(0): Comparison of Local and Nonlocal Correlation Functionals. *ACS Catal.* **2015**, *5*, 2070–2080, DOI: 10.1021/cs501675t, Publisher: American Chemical Society.

[28] Mullen, G. M.; Janik, M. J. Density Functional Theory Study of Alkane-Alkoxide Hydride Transfer in Zeolites. *ACS Catal.* **2011**, *1*, 105–115, DOI: 10.1021/cs1000619, Publisher: American Chemical Society.

[29] Ling, L. et al. General Reaction Mode of Hypervalent Iodine Trifluoromethylation Reagent: A Density Functional Theory Study. *ACS Catal.* **2015**, *5*, 2458–2468, DOI: 10.1021/cs501892s, Publisher: American Chemical Society.

[30] Hermano Sampaio Dias, A. et al. Biotransformation of Bisphenol by Human Cytochrome P450 2C9 Enzymes: A Density Functional Theory Study. *Inorg. Chem.* **2023**, *62*, 2244–2256, DOI: 10.1021/acs.inorgchem.2c03984, Publisher: American Chemical Society.

[31] Boulet, P. et al. Photochemistry of the CpNiNO Complex. A Theoretical Study Using Density Functional Theory. *Inorg. Chem.* **2001**, *40*, 7032–7039, DOI: 10.1021/ic010489c, Publisher: American Chemical Society.

[32] Vukovic, S. et al. Predicting Stability Constants for Uranyl Complexes Using Density Functional Theory. *Inorg. Chem.* **2015**, *54*, 3995–4001, DOI: 10.1021/acs.inorgchem.5b00264, Publisher: American Chemical Society.

[33] Fehér, P. P. Density Functional Theory Evaluation of a Photoinduced Intramolecular Aryl Ether Rearrangement. *J. Org. Chem.* **2021**, *86*, 2706–2713, DOI: 10.1021/acs.joc.0c02706, Publisher: American Chemical Society.

[34] Pichierri, F.; Yamamoto, Y. Mechanism and Chemoselectivity of the Pd(II)-Catalyzed Allylation of Aldehydes: A Density Functional Theory Study. *J. Org. Chem.* **2007**, *72*, 861–869, DOI: 10.1021/jo061985r, Publisher: American Chemical Society.

[35] Tian, W. Q.; Wang, Y. A. Mechanisms of Staudinger Reactions within Density Functional Theory. *J. Org. Chem.* **2004**, *69*, 4299–4308, DOI: 10.1021/jo049702n, Publisher: American Chemical Society.

[36] Liu, D.; Lin, Z. Density Functional Theory Studies of Titanium-Catalyzed Hydroboration of Olefins. *Organometallics* **2002**, *21*, 4750–4755, DOI: 10.1021/om020326b, Publisher: American Chemical Society.

[37] Ren, Q. et al. Density Functional Theory Study of the Mechanisms of Iron-Catalyzed Aminohydroxylation Reactions. *Organometallics* **2014**, *33*, 1423–1430, DOI: 10.1021/om401141r, Publisher: American Chemical Society.

[38] Witte, F. et al. Unexpected Substituent Effects in Aryl-Aryl Negishi Cross-Coupling Reactions Rationalized by Density Functional Theory and Natural Charges. *Organometallics* **2021**, *40*, 591–599, DOI: 10.1021/acs.organomet.0c00746, Publisher: American Chemical Society.

[39] Ji, M. et al. A time-dependent density functional theory study on the effect of electronic excited-state hydrogen bonding on luminescent MOFs. *Dalton Trans.* **2013**, *42*, 3464–3470, DOI: 10.1039/C2DT32575F, Publisher: The Royal Society of Chemistry.

[40] Godfrey, S. M. et al. Density functional theory studies on the reagent Ph3PBr2. *Dalton Trans.* **2005**, 1675–1678, DOI: 10.1039/B500845J, Publisher: The Royal Society of Chemistry.

[41] Bhat, C. P. et al. The role of oxygen defects in the electronic, optical and phonon dispersion of the LAGO perovskite: a density functional theory investigation. *Dalton Trans.* **2023**, *52*, 16128–16139, DOI: 10.1039/D3DT02846A, Publisher: The Royal Society of Chemistry.

[42] Chen, Y. et al. Computational insight into the mechanism and stereoselectivity of cycloaddition between donor–acceptor spirocyclopropane and aldehyde catalyzed by Brønsted acid TsOH. *Org. Biomol. Chem.* **2022**, *20*, 4006–4015, DOI: 10.1039/D2OB00140C, Publisher: The Royal Society of Chemistry.

[43] Patil, M. P.; Sunoj, R. B. Density functional theory and atoms-in-molecule study on the role of two-electron stabilizing interactions in retro Diels–Alder reaction of cycloadducts derived from substituted cyclopentadiene and p-benzoquinone. *Org. Biomol. Chem.* **2006**, *4*, 3923–3930, DOI: 10.1039/B610972A, Publisher: The Royal Society of Chemistry.

[44] Rauf, S. M. A. et al. The effect of N-methylation of amino acids (Ac-X-OMe) on solubility and conformation: a DFT study. *Org. Biomol. Chem.* **2015**, *13*, 9993–10006, DOI: 10.1039/C5OB01565K, Publisher: The Royal Society of Chemistry.

[45] Neese, F. Software update: The ORCA program system—Version 5.0. *WIREs Computational Molecular Science* **2022**, *12*, e1606, DOI: 10.1002/wcms.1606, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wcms.1606.

[46] Hanwell, M. D. et al. Avogadro: an advanced semantic chemical editor, visualization, and analysis platform. *Journal of Cheminformatics* **2012**, *4*, 17, DOI: 10.1186/1758-2946-4-17.

[47] Dennington, R. et al. GaussView Version 6. 2019; Semichem Inc. Shawnee Mission KS.

[48] Allouche, A.-R. Gabedit—A graphical user interface for computational chemistry softwares. *Journal of Computational Chemistry* **2011**, *32*, 174–182, DOI: 10.1002/jcc.21600, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jcc.21600.

[49] Otero-de-la Roza, A. et al. Critic: a new program for the topological analysis of solid-state electron densities. *Computer Physics Communications* **2009**, *180*, 157–166, DOI: 10.1016/j.cpc.2008.07.018.

[50] Otero-de-la Roza, A. et al. Critic2: A program for real-space analysis of quantum chemical interactions in solids. *Computer Physics Communications* **2014**, *185*, 1007–1018, DOI: 10.1016/j.cpc.2013.10.026.